Computer Engineering and Intelligent Systems
ISSN 2222-1719 (Paper) ISSN 2222-2863 (Online)
Vol.5, No.8, 2014

Semantic Based Automatic Question Generation using Artificial Immune System

Ibrahim Eldesoky Fattoh
Teacher Assistant at faculty of Information Technology, MUST University, Cairo, Egypt.

Abstract
This research introduces a semantic based Automatic Question Generation System using both Semantic Role Labeling and Named Entity Recognition techniques to convert the input sentence into a semantic pattern. A training phase applied to build a classifier using an Artificial Immune System that will be able to classify the patterns according to the question type. The question types considered here are set of WH-questions like who, when, where, why, and how. Then a pattern matching phase is applied to select the best matching question pattern for the test sentence. The proposed system is tested against a set of sentences obtained from different sources like Wikipedia articles, TREC 2007dataset for question answering, and English book of grade II prep. The proposed model shows promising results in determining the question type with classification accuracy increases 95%, and also in generating (matching) the new question patterns with 87%.

1. Introduction:
Natural language processing is a more popular area of artificial intelligence and it has a wide application area like text summarization, machine translation, question answering, question generation etc. [1]. Question generation is an important component in advanced learning technologies such as intelligent tutoring systems. A wide transformation has been made last years in the field of Natural Language Processing (NLP) in studying the questions as a part of the task of question answering to the task of the Automatic Question Generation. Developing Automatic Question Generation systems has been one of the important research issues because it requires insights from a variety of disciplines, including, Artificial Intelligence, Natural Language Understanding, and Natural Language Generation. One of the definitions of Question Generation is the process of automatically generating questions from various inputs like raw text, database, or semantic representation [2]. The process of AQG is divided into a set of subtasks which are content selection, question type identification, and question construction. According to the previous trials that has been applied for question generation, there are two types of question formats; multiple choice questions which asks about a word in a given sentence, the word may be an adjective, adverb, vocabulary, etc., the second format is the entity questions systems or Text to Text QG (like factual questions) which asks about a word or phrase corresponding to a particular entity in a given sentence, the question types are like what, who, why etc.. This research introduces a learning model using the Artificial Immune System (AIS) for the second type of question formats. The proposed learning model depends on the labeling roles extracted from SRL and named entities extracted from NER and the nature of the artificial immune system in supervised learning problems to build a classifier generator model for classifying the question type then generating the best matching question (s) for a given sentence. The rest of the paper is organized as follows: section 2 discusses the related work of AQG, section 3 speaks about AIS, section 4 introduces both SRL and NER , section 5 introduces the proposed model AIQGS, section 6 shows the experimental results and evaluation, and last section 7 introduces a conclusion and future work with some remarks.

2. Related work
In this section, a review of the previous AQG systems from sentence for the second question type formats mentioned in section 1 introduced. Previous efforts in QG from text can be broadly divided into three categories: syntax based, semantics based, and template based. The three categories are not entirely disjoint. In the first category, syntax based methods follow a common technique: parse the sentence to determine syntactic structure, simplify the sentence if possible, identify key phrases, and apply syntactic transformation rules and question word replacement. There are many systems in the literature for the syntax based methods. Kalady et al.[3], Varga and Ha [4], Wolfe [5], and Ali et al.[6] introduce a sample of these methods. The second category, semantic based methods also relied on transformations to generate questions from declarative sentence. They depend on the semantic analysis rather than the syntactic. Mannem et al.[7] show us a system that combines SRL with syntactic transformations to generate questions. Yao and Zhang [8] demonstrate an approach to QG based on minimal recursion semantics (MRS), a framework for shallow semantic analysis developed by Copestake et al.[9] Their method uses an eight-stage pipeline to convert input text to a set of questions. The third category, template based methods offer the ability to ask questions that are not as tightly coupled to the exact wording of the source text as syntax and semantics based methods. Cai et al.[10] presented NLGML (Natural Language Generation Markup Language), a language that can be used to generate questions of any desired type. Wang et al. [11] introduced a system to generate the questions automatically based on question templates which are

created by training the system on many medical articles. The model used in this research belongs to semantic based methods with learning phase that introduces a classifier to decide what is the question type, and a recognizer for question pattern that specify the generated question.

3. Artificial Immune System

An Artificial Immune System is a computational model based upon metaphors of the natural immune system" [12]. From information-processing perspective, the immune system is parallel and distributed adaptive system with partial decentralized control mechanism. Immune system utilizes feature extraction, learning, storage memory, and associative retrieval in order to solve recognition and classification tasks. In particular, it learns to recognize relevant patterns, remember patterns that have been seen previously, and has the ability to efficiently construct pattern detectors. The overall behavior is an emergent property of many local interactions. Such information-processing capabilities of the immune system provide many useful aspects in the field of computation [13]. AIS was emerged as a new branch in Computational Intelligence (CI) in the 1990s. A number of AIS models exist, and they are used in pattern recognition, fault detection, computer security, data analysis, scheduling, machine learning, search optimization methods and a variety of other applications researchers are exploring in the field of science and engineering [12, 13]. There are four basic algorithms in AIS which are network model, positive selection, negative selection, and clonal selection [12]. The main features of the clonal selection theory [14] are proliferation and differentiation on stimulation of cells with antigens; generation of new random genetic changes expressed subsequently as a diverse antibody pattern; and estimation of newly differentiated lymphocytes carrying low affinity antigenic receptor. The clonal selection algorithm is the one used in the learning phase of the proposed model as will be shown in section 5.

4. Semantic Role Labeling (SRL) and Named Entity Recognition (NER)

The natural language processing community has recently experienced a growth of interest in semantic roles, since they describe WHO did WHAT to WHOM, WHEN, WHERE, WHY, HOW etc. for a given situation, and contribute to the construction of meaning [15]. SRL has been used in many different applications like automatic text summarization [15] and automatic question answering [16]. Given a sentence, a semantic role labeler attempts to identify the predicates (relations and actions) and the semantic entities associated with each of those predicates. The set of semantic roles used in PropBank [17] includes both predicate-specific roles whose precise meaning are determined by their predicate and general-purpose adjunct-like modifier roles whose meaning is consistent across all predicates. The predicate specific roles are Arg0,Arg1, ..., Arg5 and ArgA. Table1 shows a complete list of the modifier roles. If we have a sentence like ***Columbus discovered America in 1492***. *The SRL parse would be as seen in (1).*

[**Columbus /Arg0] [***discovered /v:Discover] [America /Arg1] [in 1492/ ArgM-Tmp]* *(1)*

Table 1: ProbBank Arguments Roles

Role	**Meaning**
ArgM-LOC	Location
ArgM-EXT	Extent
ArgM-DIS	Discourse connectives
ArgM-ADV	Adverbial
ArgM-NEG	Negation marker
ArgM-MOD	Modal verb
ArgM-CAU	Cause
ArgM-TMP	Temporal
ArgM-PNC	Purpose
ArgM-MNR	Manner
ArgM-DIR	Direction
ArgM-PRD	Secondary prediction

Also Recognition of named entities (e.g. people, organizations, location, etc.) is an essential task in many natural language processing applications nowadays. Named entity recognition (NER) is given much attention in the research community and considerable progress has been achieved in many domains, such as newswire and biomedical [18]. If we have a sentence like ***Columbus discovered America in 1492,*** *the output of NER would be like (2).*

***[Person** Columbus] discovered [**GPE** America] in [**date**1492].* *(2)*

The attributes extracted from both NER and SRL act as the target which we search for in the sentence to identify the question type. Table 2 shows the attributes extracted; their source; and their associated question type which used in this research.

Table 2: Attributes used from SRL and NERand their question types

Target	**Source**	**Question Type**
<AM-MNR>	SRL	How
<AM-CAUS>	SRL	Why
<AM-ADV> starts with for	SRL	Why
<AM-PNC> starts with to	SRL	Why
<Person>	NER	Who
<AM-LOC>	SRL	Where
<Location>	NER	Where
<AM-TMP>	SRL	When
<Date>	NER	When
<Time>	NER	When

The next section will show how SRL and NER are used in content selection phase in the proposed model for AQG.

5. Proposed Model (AIQGS)

To generate a question, we need to perform named entity recognition, semantic role labeling, sentence and question pattern generation for the sentences used in the training, use a learning algorithm to generate a classifier able to classify the test sentence according to its question type, and use the classifier patterns to generate the question pattern for the test sentence. The overall view of the AIQGS is shown in figure 1. The input sentence is passed to feature extraction phase. In the feature extraction, the input sentence is passed to both NER system and SRL system to extract its attributes as shown in (1) and (2) in last section then constructing its pattern. After generating the sentence pattern, its question pattern (s) also is generated. Both NER and SRL of the University of ILLINOIS [19,20] were used in this research.

```
ALG_DATA_PREPERATION (Sentence)
For each sentence S in the training set
Sent_Pat←Get the sentence pattern using NER, and SRL
Set the question(s) Type(s) for the SentPat
Quest_Pat ← Build the question(s) pattern(s) for the SentPat
End for
Return the Sent_Pat and Ques_Pat.
```

After preparing the training data set as shown in ALG_DATA_PREPERATION, it is passed to learning phase. An AIS classifier generator is proposed for the training phase based on the clonal selection mechanism. The algorithm proposed is shown in ALG_TRAIN.

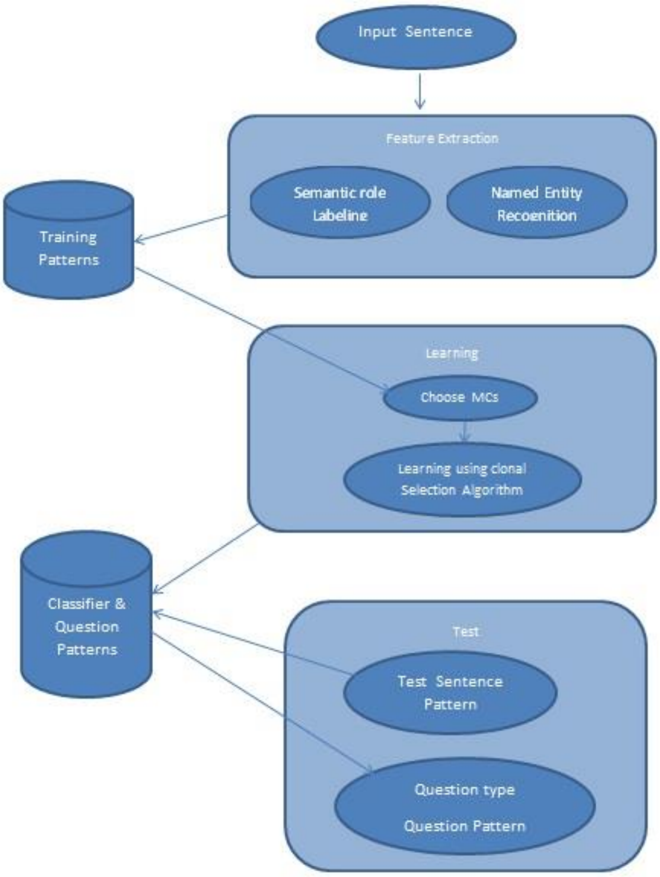


Figure 1: Flow diagram of the AQG process

The training phase starts by choosing random patterns from the training patterns to act as memory cells (MCs). Then for each antigen (AG) in the training data, the affinity between the antigen and every memory cell is measured using the Euclidean distance, then a stimulation value is calculated for each memory cell. The memory cell which has the highest stimulation value is chosen to be cloned in ALG_CLONAL _SELECTION. The stimulation value for the mutated clones also calculated and the one with the highest stimulation value is chosen to be a memory cell candidate, also if the mutated clone stimulation value is greater than the memory cell stimulation value, the memory cell is removed from the MCs pool.

```
ALG_TRAIN (Sent_Pat, Quest_Pat)
MCPool ← get random sentence and question patterns from Sent_Pat and Quest_Pat for each class.
For each Antigen (AG) in the training patterns
        Mc_Aff ← Get the affinity (AG, class MCs)
        Mc_Stim_Val ←Set the stimulation value for each MC (1-MC_Aff)
      Mc_High ← Get the MC with the highest stimulation value and its question pattern
      Mc_Clones ← CLONAL_SELECTION (Mc_High,AG,Mc_Stim_Val)
End for
```

```
ALG_CLONAL _SELECTION (MC , AG, Stim_Val)
Number_ of_Clones=Stim_Val * Clonal_Rate
For each new clone
     Clone_Aff ← Get the Affinity (clone. AG)
     Mutated_Clone ← Mutate the  clone
    Mutated_Clone_Stim_Val ← Calculate the mutated clone stimulation value    with the AG.
    Select the mutated clone with the highest stimulation value and put it in the MCs pool
    If the Mutated_Clone_Stim_Val >Stim_Val
         Remove the MC from MCs pool
End for
```

After finishing the training phase, the classifier is considered the memory cells existed in the memory cell poll, the question patterns of these memory cells are associated with them and for every mutated clone added to the memory cell pool.

The testing phase starts with preparing the new sentence as a pattern, then calculating the stimulation value for each memory cell exist in the classifier with that pattern. After that the stimulation value for each class is calculated by adding all the stimulation values of each memory cell belonging to the class. The class which has the highest average stimulation value is chosen to be the question type for the sentence. The memory cell(s) that has stimulation value greater than a given threshold value in that class is chosen to retrieve its question pattern as the question generated for the input sentence. The threshold value used here is 0.5 The generated pattern is then mapped into a sentence to form the real question of the sentence.

```
ALG_TEST (Test_Sentence)
AG ← DATA_PREPERATION  (Test_Sentence)
Foreach MC in MCsPool
    Calculate the stimulation value(MC, AG)
End for
For each class
Calculate the average stimulation value of the class for that AG
End for
Return the class with the highest average stimulation value as the question type
Get the question pattern from the MCs Pool that belongs the MC  that has stimulation value >0.5
```

5.1 Walk through example for data preparation

Applying the data preparation phase on the example shown in section 2 before, merging the output of both (1) and (2) yield a pattern like (3)

{Arg0}<person>+ {VBD} + {Arg1}<GPE>+{AM-TMP} <date> (3)

The output of the SRL is between { }, and the output of the NER is between < >. According to table 2 which show how we choose the question type according to the target extracted from NER and SRL. Searching the pattern (3); the targets Person, AM-TMP, and Date were found so two question patterns are generated for the sentence. A pattern for WHO question and the other is for WHEN question. The two patterns are

Who + {VBD} + {Arg1} <GPE> + {AM-TMP} <date>? (4)

When + did + {Arg0} <person> + {V} + {Arg1} <GPE>? (5)

The verb {V} is obtained from the output of the SRL labeling. The sentence pattern (3) and its question patterns (4) and (5) is interpreted as a two feature vectors in the training set, one vector for each question type. The representation of the feature vector is numeric data, 1, 0, and 5. One means that the attribute exists in the sentence, zero means that the attribute doesn’t exist in the sentence, five is for the attribute (s) that determines the question type.

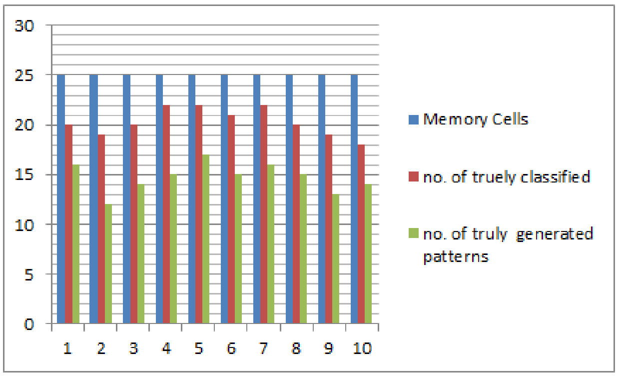
6. Experimental Results

The proposed system is applied to a set of sentences extracted from different sources like Wikipedia articles, TREC 2007dataset for question answering, and English book of grade II prep. 170 sentences are extracted and mapped into 250 patterns using SRL and NER as shown previously. The number of patterns is greater than number of sentences because some sentences are mapped into more than one pattern in case of the sentence has more than one label of either SRL or NER labels. The 250 patterns are used in training and testing. The system tested using a cross-validation test with number of folds=10, in each fold 25 patterns chosen for test. Two trials are applied to the system. In the first trial 25 patterns are randomly chosen to be memory cells in each fold. In the second trial 50 patterns are chosen as memory cells in each fold. Each time after extracting the memory cells and the test patterns, The training phase is applied to the memory cells and the remaining patterns in the training set to generate the classifier. After that the testing applied to the chosen 25 patterns. Table 3 shows the results obtained from the two trials in each fold, the number of memory cells, the number of truly classified patterns, the number of falsely classified patterns, the number of truly generated patterns for the truly classified sentences.

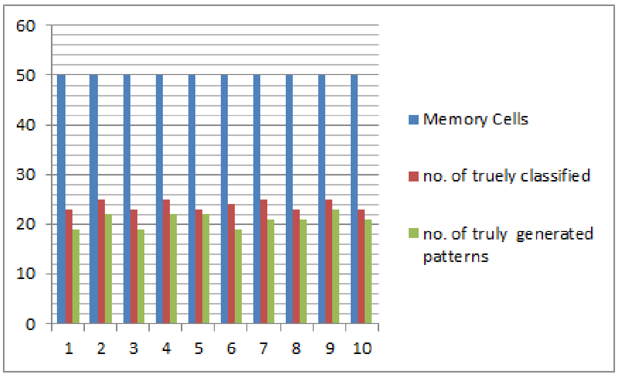
Table 3: Results of classification and question patterns generated using 25 and 50 M-cells in each test fold

EXP#	No.of M-Cells	No. of truly classified	No.of false classified	No. of truly generated patterns	Percent of true classification	Percent of true generated patterns
1	25	20	5	16	80.00%	80.00%
	50	23	2	19	92.00%	82.61%
2	25	19	6	12	76.00%	63.16%
	50	25	0	22	100.00%	88.00%
3	25	20	5	14	80.00%	70.00%
	50	23	2	19	92.00%	82.61%
4	25	22	3	15	88.00%	68.18%
	50	25	0	22	100.00%	88.00%
5	25	22	3	17	88.00%	77.27%
	50	23	2	22	92.00%	95.65%
6	25	21	4	15	84.00%	71.43%
	50	24	1	19	96.00%	79.17%
7	25	22	3	16	88.00%	72.73%
	50	25	0	21	100.00%	84.00%
8	25	20	5	15	80.00%	75.00%
	50	23	2	21	92.00%	91.30%
9	25	19	6	13	76.00%	68.42%
	50	25	0	23	100.00%	92.00%
10	25	18	7	14	72.00%	77.78%
	50	23	2	21	92.00%	91.30%

The total and average of the overall test results for the 10 folds of the two trials are shown in table 3 and figure 2. From tables 3 and 4, it is obvious that increasing the number of memory cells at the beginning of the training phase leads to the increase of the numbers of patterns in the classifier. The increasing of the number of the patterns gives the system the ability to classify more accurately and to increase the number of question patterns that have been generated or recognized truly. It is shown that out of the 10 trials, 4 times the classification accuracy reached 100% when increasing the memory cells to 50 at the beginning of the training phase. Also the percentage of the truly generated patterns increased in all trials as shown in last column in table3.



A- the 10 folds of trial 1 (25 M-Cells)



B- the 10 folds of trial 2 (50M-Cells)

Figure 2: (A) represents the 10 folds for trial 1- (B) represents the 10 folds of trial 2

The overall classification accuracy is increased by increasing the number of memory cells in the second trial from 81% to more than 95% as shown in table 4 and figure 3. The increase in the classification accuracy leads to the increase in the total truly generated patterns, as seen it jumps to 209 truly generated patterns out of 239 truly classified pattern. The increase of both total truly classified and total truly generated patterns reflects the effect of increasing the memory cells in the training process in order to generate a good classifier able to produce an accurate classification ratio.

Table 4: the total of both truly classified and generated patterns for the two trials

Trial No.	No. of M-Cells	Total truly classified	Total truly generated	Percentage of truly classified	Percentage of truly generated
1	25	203	147	81.20%	72.41%
2	50	239	209	95.60%	87.45%

The most important measures for classification problems are the precision, recall, and f-measure for each class.

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \quad (1)$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \quad (2)$$

$$F - measure = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (3)$$

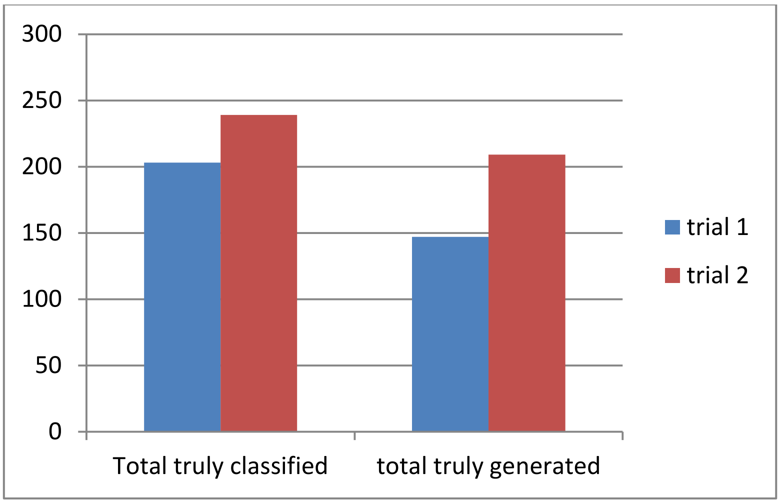


Figure 3: total truly classified and total truly generated patterns for the two trials

The precision measures the probability that the retrieved class is relevant, the recall measures the probability that a relevant class is retrieved, and the F-measure is the harmonic mean of precision and recall. These measures are shown in table 5 for each question type in this problem.

Table 5: precision, recall, and F-measure for classification of question types

Question Type	Precision	Recall	F-measure
Who	0.898	1	0.946
How	0.973	1	0.986
Why	1	0.971	0.985
When	1	0.828	0.906
Where	1	1	1

The number of truly generated patterns for each question type used in this research diver and the reason for this is the difference in the number of patterns exists in the training set for each question type. Also the memory cells are chosen randomly in each fold so sometimes the patterns chosen for a type may increase or decrease the other types. There are no common measurement units for the automatic question generation problem. Most systems use manual evaluation of experts and some other uses the measures of precision, recall, and f-measure to measure the acceptability of the system about the generated questions. In this research, because the question patterns of the test sentences are prepared, so we can use the precision, recall, and f-measure. Table 6 and figure 4 illustrates the precision, recall, and f-measure for each generated question type used in this research. The precision of the generated questions for when, Where, and Why increases 90% of their tested patterns and How and who is below this percentage, increasing the patterns for these two types may increase the percentage of generated questions for those types.

Table 6: precision, recall, and F-measure for each generated question type.

Question Type	Precision	Recall	F-Measure
Who	0.818	0.9	1
How	0.886	1	0.939
Why	0.909	0.882	0.896
When	0.917	0.772	0.838
Where	0.941	1	0.970

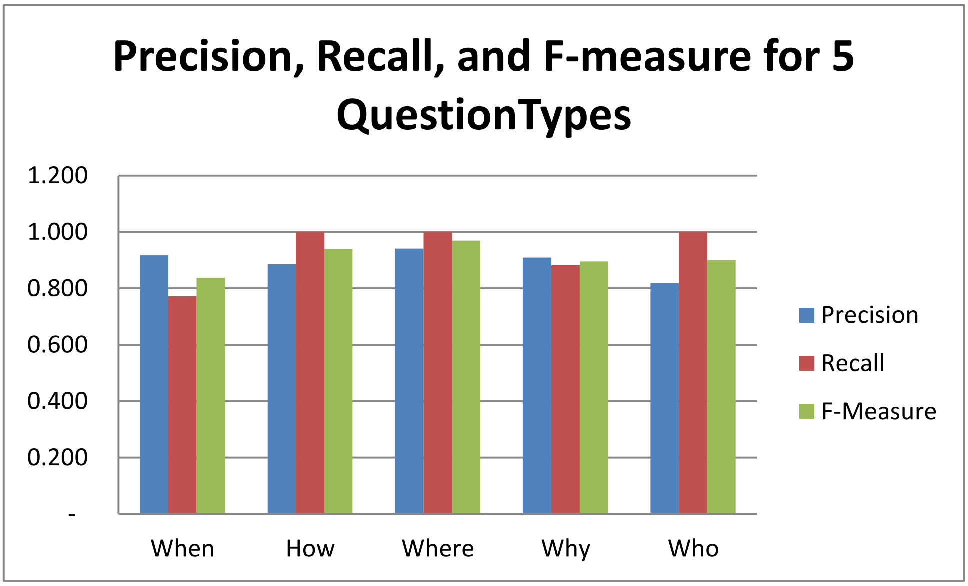


Figure 4: Precision, Recall. And F-measure for all question types

From table 6, it is shown that where, when, and why have the highest precision values and also have the highest percentage of the truly generated questions. So we can say that as the precision value increases for the question type, the percentage of the truly generated questions increase.

7. Conclusion and future work

This research introduced a novel Automatic question Generation model based on a learning model that is based on the meta-dynamics of the Artificial Immune System. The learning model depends on mapping the input sentence into a pattern, the attributes of the pattern depends on the output of two important natural language processing models; the SRL and NER are the two NLP models which builds the pattern of the sentence. The training phase generates the classifier based on the clonal selection algorithm inherited form natural immunology. The meta-dynamics of the clonal selection which apply cloning and mutation for the highly stimulated patterns leads to building a strong classifier which able to classify new patterns with accuracy greater than 95%. The classification ratio obtained proves the power of artificial immune system in multi-classes classification problem introducing the first contribution of this research which is building a learning model for identifying the question type. Preserving the question patterns during the training with the memory cells which constitutes the classifier is the idea used to generate question pattern for the new test pattern providing the second contribution of this research as a new methodology for generating the question. The percentage of truly generated patterns increased 87% which appears to be promising ratio in this problem comparing it to other techniques used in generating questions automatically. Most relative systems use a set of predefined rules that makes syntactic transformation form sentence to a question like [6 and 7]. Unfortunately the system can't be compared to other systems for different reasons, the data set used for every system is different, and the question types generated also differs from system to system. We plan to introduce a set of enhancements for the system in the future by adding into account the syntax information of the sentence beside the semantic information used. Also increasing the training sentences and question types used like what, which, how many and how long questions. Also trying to enhance the classification accuracy by hybridizing the clonal selection with genetic algorithm, increasing the classification may lead to increasing the truly generated patterns.

References:

[1] Bednarik L. and Kovacs L.. “Implementation and assessment of the automatic question generation module”. CogInfoCom 2012. 3rd IEEE International Conference on Cognitive infocommunications. December 2-5, 2012. Kosice, Solvakia.
[2] Rus V., Cai Z., and Graesser A.. “Question Generation: Example of A Multi-year Evaluation Campaign”. In Proceedings of 1st Question Generation Workshop, September 25-26, 2008, NSF, Arlington, VA.
[3] Kalady S., Elikkottil A., and Das R.. ”Natural language question generation using syntax and keywords”. In Proceedings of QG2010: The Third Workshop on Question Generation, pages 1–10, 2010.
[4] Varga A., and Ha L. “ Wlv: A question generation system for the qgstec 2010 task b” . In Proceedings of QG2010: The Third Workshop on Question Generation, pages 80–83, 2010.
[5] Wolfe J.” Automatic question generation from text-an aid to independent study”. In ACM SIGCUE Outlook, volume 10, pages 104–112. ACM, 1976.
[6] Ali H., Chali Y., and Hasan S. “ Automation of question generation from Sentences”. In Proceedings of QG2010: The Third Workshop on Question Generation, pages 58–67, 2010.
[7] Mannem P., Prasad R., and Joshi A.. “Question generation from paragraphs at upenn: Qgstec system description” . In Proceedings of QG2010: The Third Workshop on Question Generation, pages 84–91, 2010.
[8] Yao X., and Zhang Y. “Question generation with minimal recursion semantics”. In Proceedings of QG2010: The Third Workshop on Question Generation, pages 68–75. Citeseer, 2010.
[9] Copestake A., Flickinger D., Pollard D., and Sag I.” Minimal recursion semantics: An introduction”. Research on Language and Computation, 3(2-3):281–332, 2005.
[10] Cai Z., Rus V., Kim H., C.Susarla S., Karnam P., and C. Graesser. Nlgml A.” A markup language for question generation”. In Thomas Reeves and Shirley Yamashita, editors, Proceedings of World Conference on E-Learningin Corporate, Government, Healthcare, and Higher Education 2006, pages 2747–2752,Honolulu, Hawaii, USA, October 2006. AACE.
[11] Wang W., Hao T., and Liu W., “Automatic Question Generation for Learning Evaluation in Medicine “. Advances in Web Based Learning- ICWL 2007(2008), 242-251.
[12] Decastro. L. N, "Immune Cognition, Micro-evolution, and a Personal Account on Immune Engineering ". S.E.E.D. Journal (Semiotics, Evolution, Energy, and Development). Universidad de Toronto, 3(3). 2003
[13] Dasgupta D., "Advances in Artificial Immune Systems". IEEE Computational intelligence magazine, November 2006.
[14] Decastro. L. N., and Von Zuben F.J. “ Learninig and Optimization Using Clonal Selection Principle”. IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, VOL. 6, NO. 3, JUNE 2002
[15] Trandabat D. “ Using Semantic Roles to Improve Summaries”.
ENLG '11 Proceedings of the 13th European Workshop on Natural Language Generation.
pages 164-169. 2011.
[16] Pizzato L., and Molla D..”Indexing on Semantic Roles for Question Answering” . Coling 2008: Proceedings of the 2nd workshop on Information Retrieval for Question Answering (IR4QA). pages 74–81. Manchester, UK. August 2008.
[17] Palmer M., Gildea D., and Kingsbury P.. “The proposition bank: An annotated corpus of semantic roles”. Computational Linguistics, 31(1):71–106, 2005.
[18]Tkachenko M., and Simanovisky A.. “Named Entity Recognition: Exploring Features”. Proceedings of KONVENS 2012 (Main track: oral presentations), Vienna, September 20, 2012
[19] Ratinov L., and Roth D.” Design Challenges and Misconceptions in Named Entity Recognition”. CoNLL (2009)
[20] Punyakanok V. , Roth D., and Yih W. ” The Importance of Syntactic Parsing and Inference in Semantic Role Labeling”. Computational Linguistics (2008